IN THE CLAIMS:

1. (Currently amended) A method of workload management in a data processing system, comprising:

classifying a process into a class of a plurality of predefined classes, each of the plurality of predefined classes having corresponding system resource shares and ~~tiers~~ a tier;

performing workload management with respect to classes in other tiers based on priorities of the tiers; and

performing workload management with respect to other classes within a same tier as the class into which the process is classified, based on the corresponding system resource shares of the class and of the other classes, such that at least some of the processes within a same tier concurrently share system resources during execution of the at least some processes.

a'

2. (Currently amended) The method of claim 1, wherein ~~the system resource shares are used to determined~~ a percentage of system resources to be allocated to the process in the class relative to other processes in other classes is based upon the system resource shares associated with the class and the system resource shares associated with the other classes.

3. (Original) The method of claim 1, wherein the system resource tiers provide a measure of importance of a set of classes relative to another set of classes.

4. (Original) The method of claim 1, wherein the process is classified into a class based on one or more classification rules.

5. (Original) The method of claim 1, further comprising storing the process in a workload queue based on the classification of the process into the class.

6. (Original) The method of claim 1, wherein each class of the plurality of predefined classes has a minimum and a maximum resource limit, the minimum and maximum resource limits defining a minimum and a maximum amount of a resource that may be allocated to the class as a percentage of the resource.

7. (Original) The method of claim 1, wherein the system resource tiers are organized hierarchically by priority.

8. (Original) The method of claim 1, wherein performing workload management with respect to other classes within a same tier comprises determining a percentage goal for the process as a function of a number of system resource shares associated with the class in which the process is classified divided by a total number of shares allocated to active classes in the same tier as the class in which the process is classified.

9. (Original) The method of claim 1, wherein each process in each of the plurality of predefined classes has an associated priority component that is used, along with the system resource shares of the class, to calculate a resource allocation priority.

10. (Original) The method of claim 1, wherein performing workload management with respect to other classes within the same tier comprises determining a resource allocation priority range for the class based on a minimum resource usage limit, a maximum resource usage limit, and a resource usage goal.

11. (Original) The method of claim 10, wherein performing workload management with respect to other classes within the same tier further comprises adjusting the resource allocation priority range for the class based on a comparison of an actual amount of the resource being utilized by the class to the maximum resource usage limit, the minimum resource usage limit, and the resource usage goal.

12. (Original) The method of claim 8, wherein if the percentage goal is below a minimum resource usage limit, the class is favored for additional usage of the system

resource, and wherein if the calculated percentage goal is above a maximum resource usage limit, the class is not favored for additional usage of the system resource.

13. (Original) The method of claim 12, wherein the minimum resource usage limit and maximum resource usage limit are retrieved from a share/tier profile storage.

14. (Original) The method of claim 1, wherein each of the plurality of predefined classes includes an associated absolute maximum resource usage limit, and wherein if the resource allocation for processes in a class exceed an absolute maximum resource usage limit for the class, additional resource allocation to the class is suspended.

15. (Original) The method of claim 1, further comprising determining a class priority adjustment for each of the plurality of predefined classes, wherein the class priority adjustment is determined based on a delta value computed for every predetermined time increment.

16. (Original) The method of claim 15, wherein the delta value is a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment.

17. (Original) The method of claim 10, further comprising adjusting a priority of the class by a class priority adjustment value, the class priority adjustment value being determined based on a comparison of actual resource usage to the minimum resource usage limit, the maximum resource usage limit, and a resource usage goal.

18. (Original) The method of claim 17, wherein if the actual resource usage is between the maximum resource usage limit and an absolute maximum resource usage limit, the class priority adjustment is set to disfavor the class from being allocated additional amounts of the resource.

19. (Original) The method of claim 17, wherein if the actual resource usage is between the resource usage goal and the maximum resource usage limit and a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is less than zero, the class priority adjustment is not changed.

20. (Original) The method of claim 17, wherein if the actual resource usage is between the minimum resource usage limit and the resource usage goal and a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is greater than zero, the class priority adjustment is incremented by a function of the difference.

21. (Original) The method of claim 17, wherein if the actual resource usage is below the resource usage goal and the difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is greater than zero, the class priority adjustment is not changed.

22. (Original) The method of claim 17, wherein if the actual resource usage is below the resource usage goal and the difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is less than or equal to zero, the class priority adjustment is decremented by a function of the difference.

23. (Original) The method of claim 7, wherein processes in classes in lower priority tiers are only able to obtain access to a resource if processes in classes in higher priority tiers do not use all of the resource or if the processes in each of the classes in the higher priority tier have reached an absolute maximum resource usage limit.

24. (Currently amended) A computer program product in a computer readable medium for workload management in a data processing system, comprising:

first instructions for classifying a process into a class of a plurality of predefined classes, each of the plurality of predefined classes having corresponding system resource shares and ~~tiers~~ a tier;

second instructions for performing workload management with respect to classes in other tiers based on priorities of the tiers; and

third instructions for performing workload management with respect to other classes within a same tier as the class into which the process is classified, based on the corresponding system resource shares of the class and of the other classes, such that at least some of the processes within a same tier concurrently share system resources during execution of the at least some processes.

25. (Currently amended) The computer program product of claim 24, wherein ~~the system resource shares are used to determined~~ a percentage of system resources to be allocated to the process in the class relative to other processes in other classes is based upon the system resource shares associated with the class and the system resource shares associated with the other classes.

26. (Original) The computer program product of claim 24, wherein the system resource tiers provide a measure of importance of a set of classes relative to another set of classes.

27. (Original) The computer program product of claim 24, wherein the process is classified into a class based on one or more classification rules.

28. (Original) The computer program product of claim 24, further comprising fourth instructions for storing the process in a workload queue based on the classification of the process into the class.

29. (Original) The computer program product of claim 24, wherein each class of the plurality of predefined classes has a minimum and a maximum resource limit, the

minimum and maximum resource limits defining a minimum and a maximum amount of a resource that may be allocated to the class as a percentage of the resource.

30. (Original) The computer program product of claim 24, wherein the system resource tiers are organized hierarchically by priority.

31. (Original) The computer program product of claim 24, wherein the third instructions for performing workload management with respect to other classes within a same tier comprises instructions for determining a percentage goal for the process as a function of a number of system resource shares associated with the class in which the process is classified divided by a total number of shares allocated to active classes in the same tier as the class in which the process is classified.

32. (Original) The computer program product of claim 24, wherein each process in each of the plurality of predefined classes has an associated priority component that is used, along with the system resource shares of the class, to calculate a resource allocation priority.

33. (Original) The computer program product of claim 24, wherein the third instructions for performing workload management with respect to other classes within the same tier comprises instructions for determining a resource allocation priority range for the class based on a minimum resource usage limit, a maximum resource usage limit, and a resource usage goal.

34. (Original) The computer program product of claim 33, wherein the third instructions for performing workload management with respect to other classes within the same tier further comprises instructions for adjusting the resource allocation priority range for the class based on a comparison of an actual amount of the resource being utilized by the class to the maximum resource usage limit, the minimum resource usage limit, and the resource usage goal.

35. (Original) The computer program product of claim 31, wherein if the percentage goal is below a minimum resource usage limit, the class is favored for additional usage of the system resource, and wherein if the calculated percentage goal is above a maximum resource usage limit, the class is not favored for additional usage of the system resource.

36. (Original) The computer program product of claim 35, wherein the minimum resource usage limit and maximum resource usage limit are retrieved from a share/tier profile storage.

37. (Original) The computer program product of claim 24, wherein each of the plurality of predefined classes includes an associated absolute maximum resource usage limit, and wherein if the resource allocation for processes in a class exceed an absolute maximum resource usage limit for the class, additional resource allocation to the class is suspended.

38. (Original) The computer program product of claim 24, further comprising fourth instructions for determining a class priority adjustment for each of the plurality of predefined classes, wherein the class priority adjustment is determined based on a delta value computed for every predetermined time increment.

39. (Original) The computer program product of claim 38, wherein the delta value is a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment.

40. (Original) The computer program product of claim 33, further comprising fourth instructions for adjusting a priority of the class by a class priority adjustment value, the class priority adjustment value being determined based on a comparison of actual resource usage to the minimum resource usage limit, the maximum resource usage limit, and a resource usage goal.

41. (Original) The computer program product of claim 40, wherein if the actual resource usage is between the maximum resource usage limit and an absolute maximum resource usage limit, the class priority adjustment is set to disfavor the class from being allocated additional amounts of the resource.

42. (Original) The computer program product of claim 40, wherein if the actual resource usage is between the resource usage goal and the maximum resource usage limit and a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is less than zero, the class priority adjustment is not changed.

43. (Original) The computer program product of claim 40, wherein if the actual resource usage is between the minimum resource usage limit and the resource usage goal and a difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is greater than zero, the class priority adjustment is incremented by a function of the difference.

44. (Original) The computer program product of claim 40, wherein if the actual resource usage is below the resource usage goal and the difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is greater than zero, the class priority adjustment is not changed.

45. (Original) The computer program product of claim 40, wherein if the actual resource usage is below the resource usage goal and the difference between an average resource usage over a specified time interval and a resource usage for a last predetermined time increment is less than or equal to zero, the class priority adjustment is decremented by a function of the difference.

46. (Original) The computer program product of claim 30, wherein processes in classes in lower priority tiers are only able to obtain access to a resource if processes in

classes in higher priority tiers do not use all of the resource or if the processes in each of the classes in the higher priority tier have reached an absolute maximum resource usage limit.

47. (Original) A workload management apparatus in a data processing system, comprising:

a process classifier for classifying a process into a class of a plurality of predefined classes, the plurality of predefined classes having corresponding system resource shares and tiers; and

a processor coupled to the process classifier for performing workload management with respect to classes in other tiers based on priorities of the tiers, and performing workload management with respect to other classes within a same tier as the class into which the process is classified, based on the corresponding system resource shares of the class and of the other classes.

48. (Currently amended) The workload management apparatus of claim 47, wherein ~~the system resource shares are used to determined a~~ percentage of system resources to be allocated to the process in the class relative to other processes in other classes is based upon the system resource shares associated with the class and the system resource shares associated with the other classes.

49. (Original) The workload management apparatus of claim 47, wherein the system resource tiers provide a measure of importance of a set of classes relative to another set of classes.

50. (Original) The workload management apparatus of claim 47, further comprising a classification rules storage device for storing classification rules, wherein the process classifier classifies the process into a class based on one or more classification rules.